Microsoft Knowledge Base Article - 270844

HOWTO: Create an Excel Histogram by Using Automation and Analysis ToolPak

Applies To

This article was previously published under Q270844

SUMMARY

This article provides a sample Visual Basic project that shows how to use the Analysis ToolPak (ATP) to generate a histogram. This article also illustrates basic techniques for implementing and using the Analysis ToolPak add-in library, and the Analysis ToolPak Visual Basic for Applications (VBA) template add-in with Automation.

MORE INFORMATION

1. Create a new **Standard EXE** project in Visual Basic. **Form1** is created by default.
2. Add a **Command** button to **Form1**.
3. On the **Project** menu, select **References**, and then select the **Microsoft Excel Object Library**. Note that the version varies depending on the Microsoft Office version that is installed on your system.
4. Add the following code to the **Command1_Click** event:

```
''' This project requires project references to
''' The Excel object library - e.g. Excel9.olb or Excel8.olb

    Dim oXl As Excel.Application
    Dim oBook As Excel.Workbook
    Dim oSheet As Excel.Worksheet
    Dim oRange As Excel.Range
    Dim oAddIn As Excel.AddIn

    'Launch Excel and make it visible
    Set oXl = CreateObject("Excel.application")
    oXl.Visible = True
    Set oBook = oXl.Workbooks.Add
    Set oSheet = oBook.Worksheets.Item(1)

    ' Add the Excel Analysis ToolPak library
    oXl.AddIns.Add FileName:=oXl.LibraryPath & "\analysis\analys32.xll"
    Set oAddIn = oXl.AddIns.Item("Analysis ToolPak")

    ' Register all the Analysis ToolPak functions
    '   See Microsoft Knowledge Base Article Q108888, or Q213489
    oXl.RegisterXLL "Analys32.xll"

    ' Add the Excel Analysis ToolPak - VBA AddIn -
    '   it's the Automation interface to the Analysis ToolPak library
    ' Now open the .xla so that you can run its Auto_Open macro now, and others later.
    ' See Microsoft Knowledge Base article Q108888, or Q213489
    oXl.Workbooks.Open oXl.LibraryPath & "\analysis\atpvbaen.xla"
    oXl.Workbooks("atpvbaen.xla").RunAutoMacros 1

    ' Fill the worksheet with some data
    ' Create and fill the Input Range - See Microsoft Knowledge Base
    ' Article Q141684 or Q214269
    Set oRange = oSheet.Cells(1, 1)
    oRange.Value = "87"
    Set oRange = oSheet.Cells(2, 1)
    oRange.Value = "27"
```

```
                    Set oRange = oSheet.Cells(3, 1)
                    oRange.Value = "45"
                    Set oRange = oSheet.Cells(4, 1)
                    oRange.Value = "62"
                    Set oRange = oSheet.Cells(5, 1)
                    oRange.Value = "3"
                    Set oRange = oSheet.Cells(6, 1)
                    oRange.Value = "52"
                    Set oRange = oSheet.Cells(7, 1)
                    oRange.Value = "20"
                    Set oRange = oSheet.Cells(8, 1)
                    oRange.Value = "43"
                    Set oRange = oSheet.Cells(9, 1)
                    oRange.Value = "74"
                    Set oRange = oSheet.Cells(10, 1)
                    oRange.Value = "61"

                    ' Create and populate the Bin Range
                    Set oRange = oSheet.Cells(1, 2)
                    oRange.Value = "20"
                    Set oRange = oSheet.Cells(2, 2)
                    oRange.Value = "40"
                    Set oRange = oSheet.Cells(3, 2)
                    oRange.Value = "60"
                    Set oRange = oSheet.Cells(4, 2)
                    oRange.Value = "80"

                    ' Chart the Histogram on a new Worksheet
                    ' See Microsoft Knowledge Base article Q108888, or Q213489
                    oXl.Run "ATPVBAEN.XLA!Histogram", oXl.ActiveSheet.Range("$A$1:$A$10"), _
                            "", oXl.ActiveSheet.Range("$B$1:$B$4"), _
                            False, False, True, False ' True = Chart

                    'Clean up
                    Set oAddIn = Nothing
                    Set oRange = Nothing
                    Set oSheet = Nothing
                    Set oBook = Nothing
                    oXl.UserControl = True
                    Set oXl = Nothing
```

5. Run the project, and then click the **Command** button on **Form1**. Results: The histogram's input range and bin range are added to Sheet1!A1:A10 and Sheet1!B1:B4, respectively. The histogram's output range and chart are then added to a new worksheet ply in the workbook.

REFERENCES

For additional information, click the following article numbers to view the articles in the Microsoft Knowledge Base:

108888 XL: Add-Ins Don't Load When Using CreateObject

213489 XL2000: Add-Ins Don't Load When Using CreateObject Command

141684 How to Use the Histogram Add-in

214269 HOW TO: Use the Histogram Add-in in Excel 2000

For more information about Office Automation, visit the following Microsoft Web site:

http://support.microsoft.com/support/officedev/

The information in this article applies to:

- Microsoft Office Excel 2003
- Microsoft Excel 2000
- Microsoft Excel 97 for Windows
- Microsoft Visual Basic Professional Edition for Windows 6.0
- Microsoft Visual Basic Professional Edition for Windows 5.0
- Microsoft Visual Basic Enterprise Edition for Windows 6.0
- Microsoft Visual Basic Enterprise Edition for Windows 5.0

Last Reviewed: 12/12/2003 (4.0)

Keywords: kbAutomation kbhowto kbProgramming KB270844

Send   Print   Help

© 2004 Microsoft Corporation. All rights reserved. Terms of use Privacy statement Accessibility